Chapter 59
Design of Language Learning Software

Vehbi Turel
The University of Bingol, Turkey

Peter McKenna
Manchester Metropolitan University, UK

ABSTRACT

In this chapter, the authors focus on the principles and guidelines that should be borne in mind when designing and developing (interactive multimedia) language software for foreign language learning. The stages of software design and development can be categorized into six separate stages: (1) feasibility, (2) setting up a team of experts, (3) designing, (4) programming, (5) testing, and (6) evaluating. Each stage is vital to the design and development process for cost effective software, and a wide range of principles and guidelines need to be borne in mind at each stage in order to design and develop effective language learning software. Here, the authors focus on the design principles and guidelines that need to be considered while designing and developing language software.

INTRODUCTION

It would not be wrong to say that there were limitations to educational technology a decade ago, for example, the diminished quality of compressed video clips (Soboleva & Tronenko, 2002, pp. 488, 496). Software on the market used to be called 'shovelware' (Clifford, 1998, pp. 2-8; Le Mon, 1988, p. 39). Since then development in the field of educational technology has occurred very rapidly. As a result, educational technology can now enable us to design and develop technologically very highly sophisticated software for Foreign Language Learning (FLL). The main problem is no longer the technological dimension.

However, there are still many programs for FLL on the market some aspects of which are not sophisticated pedagogically and psychologically (Turel, 2010, p. 399; Draper, 2009, pp. 306-315; Trinder, 2002, pp. 69-84; Ferney & Waller, 2001, p. 156) although research in the field of language software development has increased tremendously (Hwa, et al., 2012, pp. 35-50; Abobaker & Hussein, 2012, pp. 61-63; Godwin-Jones, 2010; Blake,

DOI: 10.4018/978-1-4666-4301-7.ch059

2008; Zaphiris, 2006). Some programs even feature spelling errors although written by native speakers (TES Teacher, 2004, p. 18).

In short, problems now fundamentally stem from materials writers, not the technology itself. Therefore, to be able to design and develop cost effective and professional software for FLL, there are certain scientific educational findings and implications that need to be implemented at every single stage of program design and development (Draper, 2009; Turel, 2004; Brett, 1999; Peter, 1994).

In order for us to be able to create sophisticated language software, teams of experts are needed (Turel, 2004, p. 140). Under normal conditions, to be able to create cost-effective software for FLL, the active participation of most of the experts below (depending on the type of the language software program we want to create) is essential. These are: (specialist language) teachers/experts, programmers, graphic designers, audio engineers, photographers, artists, voice actors, film directors/specialists, musicians, animators, and (the target) Language Learners (LLs) (see Figure 1).

The involvement of these experts is necessary and important (Nicholson & Ngai, 1996, p. 3). For instance, target LLs' involvement, 'produces more useable and effective' software (Nikolova, 2002, p. 112; Kennedy & McNaught, 1997, p. 6; MacGregor, 1993, pp. 61, 63; Eraut, 1988), although all materials that are based on findings practically, to some extent, feature LLs' involvement indirectly, as the findings are very often LLs' preferences, views, ideas, progress etc. Likewise, the lack of a specialist programmer hinders not only the use of the maximum potential of the tools, but also results in a lack of the minimum requirements. In one project, for instance, it came to a point where the developers had to ask a 'programming specialist to take over … the software development work' (Grob & Wolff, 2001, p. 249). Similarly, in Lyall and McNamara's (2000, p. 133) case, due to technical problems the LLs encountered, many LLs did not continue studying with the CALL program. In Debski and Gruba's (1999, pp. 219-239) study, the problems in the computer classroom were mostly technology related and the software was seen as unreliable

Figure 1. Potential experts that are essential for creating real sense cost effective language learning software